A Rough Set Approach to Data with Missing Attribute Values

Jerzy W. Grzymala-Busse

Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS 66045, USA
and
Institute of Computer Science Polish Academy of Sciences, 01-237 Warsaw, Poland
jerzy@ku.edu
http://lightning.eecs.ku.edu/index.html

Abstract. In this paper we discuss four kinds of missing attribute values: lost values (the values that were recorded but currently are unavailable), "do not care" conditions (the original values were irrelevant), restricted "do not care" conditions (similar to ordinary "do not care" conditions but interpreted differently, these missing attribute values may occur when in the same data set there are lost values and "do not care" conditions), and attribute-concept values (these missing attribute values may be replaced by any attribute value limited to the same concept). Through the entire paper the same calculus, based on computations of blocks of attribute-value pairs, is used. Incomplete data are characterized by characteristic relations, which in general are neither symmetric nor transitive. Lower and upper approximations are generalized for data with missing attribute values. Finally, some experiments on different interpretations of missing attribute values and different approximation definitions are cited.

Keywords: Incomplete data sets, lost values, "do not care" conditions, attribute-concept values, blocks of attribute-value pairs, characteristic sets, characteristic relations.

1 Introduction

Using standard rough set theory we may describe complete data sets, i.e., data sets in which all attribute values are given. However, many real-life data sets are incomplete, i.e., some attribute values are missing. Recently rough set theory was extended to handle incomplete data sets (i.e., data sets with missing attribute values) [1,2,3,4,5,6,7,8,9,10,11,12,13,17,18,19].

We will consider four kinds of missing attribute values [7]. The first kind of missing attribute value will be called *lost*. A missing attribute value is lost when for some case (example, object) the corresponding attribute value was mistakenly erased or the entry into the data set was forgotten. In these cases the original value existed, but for a variety of reasons now it is not accessible.

The next three kinds of missing attribute values, called *"do not care" conditions*, *restricted "do not care" conditions* and *attribute-concept values* are based

G. Wang et al. (Eds.): RSKT 2006, LNAI 4062, pp. 58–67, 2006.

on an assumption that these values were initially, when the data set was created, irrelevant. For example [7], in a medical setup, patients were subjected to preliminary tests. Patients whose preliminary test results were negative were diagnosed as not affected by a disease. They were perfectly well diagnosed in spite of the fact that not all tests were conducted on them. Thus some test results are missing because these tests were redundant. In different words, a missing attribute value of this kind may be potentially replaced by any value typical for that attribute. This kind of a missing attribute value will be called a "do not care" condition. Restricted "do not care" conditions are defined in the next section. In our last case, a missing attribute value may be replaced by any attribute value limited to the same concept. For example [7], if a patient was diagnosed as not affected by a disease, we may want to replace the missing test (attribute) value by any typical value for that attribute but restricted to patients in the same class (concept), i.e., for other patients not affected by the disease. Such missing attribute value will be called attribute-concept value.

Two special data sets with missing attribute values were extensively studied: in the first case, all missing attribute values are lost, in the second case, all missing attribute values are ordinary "do not care" conditions. Incomplete decision tables in which all attribute values are lost, from the viewpoint of rough set theory, were studied for the first time in [10], where two algorithms for rule induction, modified to handle lost attribute values, were presented. This approach was studied later, e.g., in [17,18], where the indiscernibility relation was generalized to describe such incomplete decision tables.

On the other hand, incomplete decision tables in which all missing attribute values are "do not care" conditions, from the view point of rough set theory, were studied for the first time in [2], where a method for rule induction was introduced in which each missing attribute value was replaced by all values from the domain of the attribute. Originally [2] such values were replaced by all values from the entire domain of the attribute, later [8], by attribute values restricted to the same concept to which a case with a missing attribute value belongs. Such incomplete decision tables, with all missing attribute values being "do not care conditions", were broadly studied in [12,13], including extending the idea of the indiscernibility relation to describe such incomplete decision tables.

In general, incomplete decision tables are described by characteristic relations, in a similar way as complete decision tables are described by indiscernibility relations [4,5,6,7].

In rough set theory, one of the basic notions is the idea of lower and upper approximations. For complete decision tables, once the indiscernibility relation is fixed and the concept (a set of cases) is given, the lower and upper approximations are unique.

For incomplete decision tables, for a given characteristic relation and concept, there are three important and different possibilities to define lower and upper approximations, called singleton, subset, and concept approximations [4,5,6,7]. Singleton lower and upper approximations were studied in [12,13,16,17,18]. Note that similar three definitions of lower and upper approximations, though not for incomplete decision tables, were studied in [14,20]. As it was observed in [4,5,6,7], singleton lower and upper approximations are not applicable in data mining.

2 Blocks of Attribute-Value Pairs

We assume that the input data sets are presented in the form of a *decision table.* An example of a decision table is shown in Table 1. Rows of the decision table

Table 1. An incomplete decision table

	Attributes			Decision
Case	Capacity	Noise	Size	Quality
1	two	–	compact	high
2	four	*	*	high
3	?	medium	medium	low
4	+	low	compact	low
5	four	?	medium	high
6	–	medium	full	low
7	five	low	full	high

represent *cases*, while columns are labeled by *variables.* The set of all cases will be denoted by U. In Table 1, $U = \{1, 2, ..., 7\}$. Independent variables are called *attributes* and a dependent variable is called a *decision* and is denoted by d. The set of all attributes will be denoted by A. In Table 1, $A = \{$*Capacity*, *Noise*, *Size*$\}$. Any decision table defines a function ρ that maps the direct product of U and A into the set of all values. For example, in Table 1, $\rho(1, Capacity) = two$. A decision table with an incompletely specified function ρ will be called *incomplete.*

For the rest of the paper we will assume that all decision values are specified, i.e., they are not missing. Also, we will assume that lost values will be denoted by "?", "do not care" conditions by "*", restricted "do not care" conditions by "+", and attribute-concept values by "−". Additionally, we will assume that for each case at least one attribute value is specified.

An important tool to analyze complete decision tables is a block of the attribute-value pair. Let a be an attribute, i.e., $a \in A$ and let v be a value of a for some case. For complete decision tables if $t = (a, v)$ is an attribute-value pair then a *block* of t, denoted $[t]$, is a set of all cases from U that for attribute a have value v. For incomplete decision tables the definition of a block of an attribute-value pair must be modified in the following way:

- If for an attribute a there exists a case x such that $\rho(x, a) =?$, i.e., the corresponding value is lost, then the case x should not be included in any blocks$[(a, v)]$ for all values v of attribute a,

– If for an attribute a there exists a case x such that the corresponding value is a "do not care" condition or a restricted "do not care" condition, i.e., $\rho(x, a) = *$ or $\rho(x, a) = +$, then the case x should be included in blocks $[(a, v)]$ for all specified values v of attribute a,
– If for an attribute a there exists a case x such that the corresponding value is an attribute-concept value, i.e., $\rho(x, a) = -$, then the corresponding case x should be included in blocks $[(a, v)]$ for all specified values $v \in V(x, a)$ of attribute a, where

$$V(x, a) = \{\rho(y, a) \mid \rho(y, a) \text{ is specified}, \; y \in U, \; \rho(y, d) = \rho(x, d)\}.$$

In Table 1, for case 1, $\rho(1, Noise) = -$, and $V(1, Noise) = \{low\}$, so we add case 1 to [(*Noise*, *low*)]. For case 2, $\rho(2, Temperature) = *$, hence case 2 is included in both sets: $[(Noise,\ medium)]$ and $[(Noise,\ low)]$. Similarly, $\rho(2, Size) = *$, hence case 2 is included in all three sets: $[(Size, compact)]$, $[(Size, medium)]$, and $[(Size, full)]$.

Furthermore, ρ(3, *Headache*) = ?, so case 3 is not included in [(*Capacity*, *two*)], [(*Capacity*, *four*)] and [(*Capacity*, *five*)]. For case 4, ρ(4, *Capacity*) = +, so case 4 is included in [(*Capacity*, *two*)], [(*Capacity*, *four*)], and [(*Capacity*, *five*)]. Also, $\rho(5, Noise) = ?$, so case 5 is not in $[(Noise,\ medium)]$ and $[(Noise,\ low)]$. Finally, ρ(6, *Capacity*) = −, and $V(6, Capacity) = \emptyset$ so case 6 is not included in [(*Capacity*, *two*)], [(*Capacity*, *four*)], and [(*Capacity*, *five*)]. Thus,

[(Capacity, two)] = {1, 4},
[(Capacity, four)] = {2, 4, 5},
[(Capacity, five)] = {4, 7},
[(Noise, medium)] = {2, 3, 6},
[(Noise, low)] = {1, 2, 4, 7},
[(Size, compact)] = {1, 2, 4},
[(Size, medium)] = {2, 3, 5},
[(Size, full)] = {2, 6, 7}.

For a case $x \in U$ the *characteristic set* $K_B(x)$ is defined as the intersection of the sets $K(x, a)$, for all $a \in B$, where the set $K(x, a)$ is defined in the following way:

– If $\rho(x, a)$ is specified, then $K(x, a)$ is the block $[(a, \rho(x, a)]$ of attribute a and its value $\rho(x, a)$,
– If $\rho(x, a) = ?$ or $\rho(x, a) = *$ then the set $K(x, a) = U$,
– If $\rho(x, a) = +$, then $K(x, a)$ is equal to the union of all blocks of (a, v), for all specified values v of attribute a,
– If $\rho(x, a) = -$, then the corresponding set $K(x, a)$ is equal to the union of all blocks of attribute-value pairs (a, v), where $v \in V(x, a)$ if $V(x, a)$ is nonempty. If $V(x, a)$ is empty, $K(x, a) = U$.

For Table 1 and $B = A$,

$K_A(1) = \{1, 4\} \cap \{1, 2, 4, 7\} \cap \{1, 2, 4\} = \{1, 4\}$,
$K_A(2) = \{2, 4, 5\} \cap U \cap U = \{2, 4, 5\}$,

$K_A(3) = U \cap \{2, 3, 6\} \cap \{2, 3, 5\} = \{2, 3\}$,
$K_A(4) = (\{1, 4\} \cup \{2, 4, 5\} \cup \{4, 7\}) \cap \{1, 2, 4, 7\} \cap \{1, 2, 4\} = \{1, 2, 4\}$,
$K_A(5) = \{2, 4, 5\} \cap U \cap \{2, 3, 5\} = \{2, 5\}$,
$K_A(6) = U \cap \{2, 3, 6\} \cap \{2, 6, 7\} = \{2, 6\}$, and
$K_A(7) = \{4, 7\} \cap \{1, 2, 4, 7\} \cap \{2, 6, 7\} = \{7\}$.

Characteristic set $K_B(x)$ may be interpreted as the set of cases that are indistinguishable from x using all attributes from B and using a given interpretation of missing attribute values. Thus, $K_A(x)$ is the set of all cases that cannot be distinguished from x using all attributes.

The characteristic relation $R(B)$ is a relation on U defined for $x, y \in U$ as follows

$$(x, y) \in R(B) \textit{ if and only if } y \in K_B(x).$$

Thus, the relation $R(B)$ may be defined by $(x, y) \in R(B)$ if and only if y is indistinguishable from x by all attributes from B. The characteristic relation $R(B)$ is reflexive but—in general—does not need to be symmetric or transitive. Also, the characteristic relation $R(B)$ is known if we know characteristic sets $K(x)$ for all $x \in U$. In our example, $R(A) = \{(1, 1), (1, 4), (2, 2), (2, 4), (2, 5), (3, 2), (3, 3), (4, 1), (4, 2), (4, 4), (5, 2), (5, 5), (6, 2), (6, 6), (7, 7)\}$. The most convenient way is to define the characteristic relation through the characteristic sets.

For decision tables, in which all missing attribute values are lost, a special characteristic relation was defined in [17], see also, e.g., [18]. For decision tables where all missing attribute values are "do not care" conditions a special characteristic relation was defined in [12], see also, e.g., [13]. For a completely specified decision table, the characteristic relation $R(B)$ is reduced to the indiscernibility relation.

3 Definability

For completely specified decision tables, any union of elementary sets of B is called a B-definable set [15]. Definability for completely specified decision tables should be modified to fit into incomplete decision tables. For incomplete decision tables, a union of some intersections of attribute-value pair blocks, where such attributes are members of B and are distinct, will be called B-*locally definable* sets. A union of characteristic sets $K_B(x)$, where $x \in X \subseteq U$ will be called a B-*globally definable* set. Any set X that is B-globally definable is B-locally definable, the converse is not true. For example, the set $\{4\}$ is A-locally definable since $\{4\} = [(Capacity, five)] \cap [(Size, compact)]$. However, the set $\{4\}$ is not A-globally definable. Obviously, if a set is not B-locally definable then it cannot be expressed by rule sets using attributes from B. This is why it is so important to distinguish between B-locally definable sets and those that are not B-locally definable.

4 Lower and Upper Approximations

For completely specified decision tables lower and upper approximations are defined on the basis of the indiscernibility relation. Let X be any subset of the set U of all cases. The set X is called a *concept* and is usually defined as the set of all cases defined by a specific value of the decision. In general, X is not a B-definable set. However, set X may be approximated by two B-definable sets, the first one is called a *B-lower approximation* of X, denoted by $\underline{B}X$ and defined as follows

$$\{x \in U \mid [x]_B \subseteq X\}.$$

The second set is called a *B-upper approximation* of X, denoted by $\overline{B}X$ and defined as follows

$$\{x \in U \mid [x]_B \cap X \neq \emptyset\}.$$

The above shown way of computing lower and upper approximations, by constructing these approximations from singletons x, will be called the *first method.* The B-lower approximation of X is the greatest B-definable set, contained in X. The B-upper approximation of X is the smallest B-definable set containing X.

As it was observed in [15], for complete decision tables we may use a *second method* to define the B-lower approximation of X, by the following formula

$$\cup\{[x]_B \mid x \in U, [x]_B \subseteq X\},$$

and the B-upper approximation of x may be defined, using the second method, by

$$\cup\{[x]_B \mid x \in U, [x]_B \cap X \neq \emptyset\}.$$

Obviously, for complete decision tables both methods result in the same respective sets, i.e., corresponding lower approximations are identical, and so are upper approximations.

For incomplete decision tables lower and upper approximations may be defined in a few different ways. In this paper we suggest three different definitions of lower and upper approximations for incomplete decision tables. Again, let X be a concept, let B be a subset of the set A of all attributes, and let $R(B)$ be the characteristic relation of the incomplete decision table with characteristic sets $K(x)$, where $x \in U$. Our first definition uses a similar idea as in the previous articles on incomplete decision tables [12,13,17,18], i.e., lower and upper approximations are sets of singletons from the universe U satisfying some properties. Thus, lower and upper approximations are defined by analogy with the above first method, by constructing both sets from singletons. We will call these approximations *singleton.* A singleton B-lower approximation of X is defined as follows:

$$\underline{B}X = \{x \in U \mid K_B(x) \subseteq X\}.$$

A singleton B-upper approximation of X is

$$\overline{B}X = \{x \in U \mid K_B(x) \cap X \neq \emptyset\}.$$

In our example of the decision table presented in Table 1 let us say that $B = A$. Then the singleton A-lower and A-upper approximations of the two concepts: {1, 2, 4, 8} and {3, 5, 6, 7} are:

$$\underline{A}\{1, 2, 5, 7\} = \{5, 7\},$$

$$\underline{A}\{3,4,6\} = \emptyset,$$
$$\overline{A}\{1,2,5,7\} = U,$$
$$\overline{A}\{3,4,6\} = \{1,2,3,4,6\}.$$

We may easily observe that the set $\{5,7\} = \underline{A}\{1,2,5,7\}$ is not A-locally definable since in all blocks of attribute-value pairs cases 2 and 5 are inseparable. Thus, as it was observed in, e.g., [4,5,6,7], singleton approximations should not be used, in general, for data mining and, in particular, for rule induction.

The second method of defining lower and upper approximations for complete decision tables uses another idea: lower and upper approximations are unions of elementary sets, subsets of U. Therefore we may define lower and upper approximations for incomplete decision tables by analogy with the second method, using characteristic sets instead of elementary sets. There are two ways to do this. Using the first way, a *subset* B-lower approximation of X is defined as follows:

$$\underline{B}X = \cup\{K_B(x) \mid x \in U, K_B(x) \subseteq X\}.$$

A *subset* B-upper approximation of X is

$$\overline{B}X = \cup\{K_B(x) \mid x \in U, K_B(x) \cap X \neq \emptyset\}.$$

Since any characteristic relation $R(B)$ is reflexive, for any concept X, singleton B-lower and B-upper approximations of X are subsets of the subset B-lower and B-upper approximations of X, respectively. For the same decision table, presented in Table 1, the subset A-lower and A-upper approximations are

$$\underline{A}\{1,2,5,7\} = \{2,5,7\},$$
$$\underline{A}\{3,4,6\} = \emptyset,$$
$$\overline{A}\{1,2,5,7\} = U,$$
$$\overline{A}\{3,4,6\} = \{1,2,3,4,5,6\}.$$

The second possibility is to modify the subset definition of lower and upper approximation by replacing the universe U from the subset definition by a concept X. A *concept* B-lower approximation of the concept X is defined as follows:

$$\underline{B}X = \cup\{K_B(x) \mid x \in X, K_B(x) \subseteq X\}.$$

Obviously, the subset B-lower approximation of X is the same set as the concept B-lower approximation of X. A concept B-upper approximation of the concept X is defined as follows:

$$\overline{B}X = \cup\{K_B(x) \mid x \in X, K_B(x) \cap X \neq \emptyset\} = \\ = \cup\{K_B(x) \mid x \in X\}.$$

The concept upper approximations were defined in [14] and [16] as well. The concept B-upper approximation of X is a subset of the subset B-upper approximation of X. Besides, the concept B-upper approximations are truly the smallest

B-definable sets containing X. For the decision table presented in Table 1, the concept A-upper approximations are

$$\overline{A}\{1,2,5,7\} = \{1,2,4,5,7\},$$

$$\overline{A}\{3,4,6\} = \{1,2,3,4,6\}.$$

Note that for complete decision tables, all three definitions of lower approximations, singleton, subset and concept, coalesce to the same definition. Also, for complete decision tables, all three definitions of upper approximations coalesce to the same definition. This is not true for incomplete decision tables, as our example shows.

5 Results of Experiments

In Table 2 results of experiments [9] on seven well-known data sets from the UCI Machine Learning Repository are cited. Error rates were computed using ten-fold cross validation, with exception of the *echocardiogram* data set where leave-one-out was used.

Table 2. Error rates for data with missing attribute values

Data set	Missing attribute values interpreted as		
	lost values	"do not care" conditions	attribute-concept values
Breast_cancer	32.17	33.57	33.57
Echocardigram	32.43	31.08	31.08
Hepatitis	17.42	18.71	19.35
Horse	19.84	27.99	32.61
House	5.07	7.60	7.60
Soybean	12.38	20.52	16.94
Tumor	70.50	68.44	66.37

In our experiments we used the MLEM2 (Modified Learning from Examples Module, version 2) rule induction algorithm [3]. MLEM2, a modified version of the LEM2 algorithm, is a part of the LERS (Learning from Examples based on Rough Sets) data mining system. LERS computes lower and upper approximations for all concepts. Rules induced from the lower approximations are called *certain*, while rules induced from the upper approximations are called *possible*. All error rates, reported in Table 2, were computed using certain rule sets.

6 Conclusions

Four standard interpretations of missing attribute values are discussed in this paper. These interpretations may be applied to any kind of incomplete data set.

This paper shows how to compute blocks of attribute-value pairs for data sets with missing attribute values, how to compute characteristic sets (i.e., generalization of elementary sets), how to compute characteristic relations (i.e., generalization of an indiscernibility relations), and three kinds of approximations (reduced for ordinary approximations for complete data sets). Finally, results of experiments on seven data sets indicate that there is no universally best interpretation of missing attribute values.

References

1. Greco, S., Matarazzo, B., and Slowinski, R.: Dealing with missing data in rough set analysis of multi-attribute and multi-criteria decision problems. In: Zanakis, S.H., Doukidis, G., and Zopounidis, Z., Eds., *Decision Making: Recent Developments and Worldwide Applications*. Kluwer Academic Publishers, Dordrecht, Boston, London (2000) 295–316.
2. Grzymala-Busse, J.W.: On the unknown attribute values in learning from examples. In: Proceedings of the ISMIS-91, 6th International Symposium on Methodologies for Intelligent Systems, Charlotte, North Carolina (1991) 368–377.
3. Grzymala-Busse., J.W.: MLEM2: A new algorithm for rule induction from imperfect data. In: Proceedings of the IPMU'2002, 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Annecy, France (2002) 243–250.
4. Grzymala-Busse, J.W.: Rough set strategies to data with missing attribute values. In: Workshop Notes, Foundations and New Directions of Data Mining, the 3-rd International Conference on Data Mining, Melbourne, Florida (2003) 56–63.
5. Grzymala-Busse, J.W.: Data with missing attribute values: Generalization of idiscernibility relation and rule induction. *Transactions on Rough Sets*, Lecture Notes in Computer Science Journal Subline, Springer-Verlag **1** (2004) 78–95.
6. Grzymala-Busse, J.W.: Characteristic relations for incomplete data: A generalization of the indiscernibility relation. In: Proceedings of the RSCTC'2004, Fourth International Conference on Rough Sets and Current Trends in Computing, Uppsala, Sweden (2004) 244–253.
7. Grzymala-Busse, J.W.: Incomplete data and generalization of indiscernibility relation, definability, and approximations. In: Proceedings of the RSFDGrC'2005, Tenth International Conference on Rough Sets, Fuzzy Sets, data Mining, and Granular Computing, Springer-Verlag, Regina, Canada, (2005) 244–253.
8. Grzymala-Busse, J.W. and Hu, M.: A comparison of several approaches to missing attribute values in data mining. In: Proceedings of the Second International Conference on Rough Sets and Current Trends in Computing RSCTC'2000, Banff, Canada (2000) 340–347.
9. Grzymala-Busse, J.W. and Santoso, S.: Experiments on data with three interpretations of missing attribute valuesA rough set approach. Accepted for the IIS'2006 Conference, Intelligent Information Systems, New Trends in Intelligent Information Processing and WEB Mining, Ustron, Poland (2006).
10. Grzymala-Busse, J.W. and Wang A.Y.: Modified algorithms LEM1 and LEM2 for rule induction from data with missing attribute values. In: Proceedings of the Fifth International Workshop on Rough Sets and Soft Computing (RSSC'97) at the Third Joint Conference on Information Sciences (JCIS'97), Research Triangle Park, North Carolina (1997) 69–72.

11. Hong, T.P., Tseng L.H. and Chien, B.C.: Learning coverage rules from incomplete data based on rough sets. In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Hague, the Netherlands (2004) 3226–3231.
12. Kryszkiewicz, M.: Rough set approach to incomplete information systems. In: Proceedings of the Second Annual Joint Conference on Information Sciences, Wrightsville Beach, North Carolina (1995) 194–197.
13. Kryszkiewicz, M.: Rules in incomplete information systems. *Information Sciences* **113** (1999) 271–292.
14. Lin, T.Y.: Topological and fuzzy rough sets. In: Slowinski R., Ed., *Intelligent Decision Support. Handbook of Applications and Advances of the Rough Set Theory.* Kluwer Academic Publishers, Dordrecht, Boston, London (1992) 287–304.
15. Pawlak, Z.: *Rough Sets. Theoretical Aspects of Reasoning about Data.* Kluwer Academic Publishers, Dordrecht, Boston, London (1991).
16. Slowinski, R. and Vanderpooten, D.: A generalized definition of rough approximations based on similarity. *IEEE Transactions on Knowledge and Data Engineering* **12** (2000) 331–336.
17. Stefanowski, J. and Tsoukias, A.: On the extension of rough sets under incomplete information. In: Proceedings of the 7th International Workshop on New Directions in Rough Sets, Data Mining, and Granular-Soft Computing, RSFDGrC'1999, Ube, Yamaguchi, Japan (1999) 73–81.
18. Stefanowski, J. and Tsoukias, A.: Incomplete information tables and rough classification. *Computational Intelligence* **17** (2001) 545–566.
19. Wang, G.: Extension of rough set under incomplete information systems. In: Proceedings of the FUZZ_IEEE'2002, IEEE International Conference on Fuzzy Systems, vol. 2, Honolulu, Hawaii (2002) 1098–1103.
20. Yao, Y.Y.: Two views of the theory of rough sets in finite universes. *International J. of Approximate Reasoning* **15** (1996) 291–317.